Task Allocation and Motion Coordination of Multiple Autonomous Vehicles

- With application in automated container terminals

by

Asela K. Kulatunga

A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy

University of Technology, Sydney
Faculty of Engineering

August, 2008

CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirement for a degree except as fully acknowledged within the text.

I certify that the thesis has been written by me. Any help that I have received in my research work and preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Candidate

(Asela K. Kulatunga)

Sydney, August 2008

ABSTRACT

This thesis focuses on developing an approach to solve the complex problem of task allocation and motion coordination simultaneously for a large fleet of autonomous vehicles in highly constrained operational environments. The multi-vehicle task allocation and motion coordination problem consists of allocating different tasks to different autonomous vehicles and intelligently coordinating motions of the vehicles without human interaction. The motion coordination itself comprises two sub-problems: path planning and collision / deadlock avoidance. Although a number of research studies have attempted to solve one or two aspects of this problem, it is rare to note that many have attempted to solve the task allocation, path planning and collision avoidance simultaneously. Therefore, it cannot be conclusively said that, optimal or near-optimal solutions generated based on one aspect of the problem will be optimal or near optimal results for the whole problem. It is advisable to solve the problem as one complete problem rather than decomposing it. This thesis intends to solve the complex task allocation, path planning and collision avoidance problem simultaneously.

A Simultaneous Task Allocation and Motion Coordination (STAMC) approach is developed to solve the multi-vehicle task allocation and motion coordination problem in a concurrent manner. Further, a novel algorithm called Simultaneous Path and Motion Planning (SiPaMoP) is proposed for collision free motion coordination. The main objective of this algorithm is to generate collision free paths for autonomous vehicles, once they are assigned with tasks in a conventional path topology of a material handling environment. The Dijkstra and A * shortest path search algorithms are utilised in the proposed Simultaneous Path and Motion Planning algorithm.

The multi-vehicle task allocation and motion coordination problem is first studied in a static environment where all the tasks, vehicles and operating environment information are assumed to be known. The multi-vehicle task allocation and motion coordination problem in a dynamic environment, where tasks, vehicles and operating environment change with time is then investigated. Furthermore, issues like vehicle breakdowns, which are common in real world situations, are considered. The computational cost of solving the multi-vehicle STAMC problem is also

addressed by proposing a distributed computational architecture and implementing that architecture in a cluster computing system. Finally, the proposed algorithms are tested in a case study in an automated container terminal environment with a large fleet of autonomous straddle carriers.

Since the multi-vehicle task allocation and motion coordination is an NP-hard problem, it is almost impossible to find out the optimal solutions within a reasonable time frame. Therefore, this research focuses on investigating the appropriateness of heuristic and evolutionary algorithms for solving the STAMC problem. The Simulated Annealing algorithm, Ant Colony and Auction algorithms have been investigated. Commonly used dispatching rules such as first come first served, and closest task first have also been applied for comparison. Simulation tests of the proposed approach is conducted based on information from the Fishermen Island's container terminal of Patrick Corporation (Pty.) Ltd in Queensland, Australia where a large fleet of autonomous straddle carriers operate. The results shows that the proposed meta-heuristic techniques based simultaneous task allocation and motion coordination approach can effectively solve the complex multi-vehicle task allocation and motion coordination problem and it is capable of generating near optimal results within an acceptable time frame.

ACKNOWLEDGEMENT

This PhD thesis could not have been successfully completed if not for the tireless assistance and guidance with understanding of a list of academics throughout my PhD candidature. It is with a high sense of gratitude that I wish to place on record, the valuable guidance of my principal supervisor that I received throughout. I would be failing in my duties if I do not purposely thank Professor Dikai Liu who has given me valuable advice and encouragement, with understanding throughout the course of study. A special word of thanks should necessarily go to Professor Gamini Dissanayake for his support and guidance given throughout my candidature which paved the way for me to realize my dream of successfully completing a PhD in one of the leading Autonomous research groups.

I greatly acknowledge the financial assistance provided by the ARC Centre of Excellence for Autonomous Systems at the University of Technology, Sydney, Australia; the Faculty of Engineering, University of Technology, Sydney; and the Presidential Fund of Sri Lanka for its financial support. Furthermore, I would like to express my gratitude to Professor Sarath Siyambalapitiya of the Faculty of Engineering, University of Peradeniya, Sri Lanka for all the support given as an alternative supervisor during my stay in Sri Lanka. I would like to convey my gratitude to Doctors Peter Wu, Brad Skinner, Haye Lau and Raymond Kwok for their kind assistance and support.

I wish to express my sincere gratitude also to Dr. Fook Choon Choi and Ms. Pui Yeng Wong for their whole hearted parental support rendered with understanding during the difficult times that I had to encounter. My sincere thanks also go to Ms. Iroshini Gunaratne, Ms. Erandi Wattegama and Mr. Mohan Samaranayake for their support given to me in formatting and writing the thesis. The author wishes to express his gratitude to the friends both at UTS and elsewhere for their support and encouragement. I would fail in my duty if I do not thank Mr. Janitha Wijesinghe for keeping company with me and often providing transport to travel in and around Sydney. Furthermore, I would like to thank Mr. Tonmoy Dutta-Roy, Mr. Bashar Ramadin, and Mr. Ashod Donikian who accompanied me in my stay in Australia. My acknowledgements also are extended to CAS research students, Mr.Mashall Yuan and Mr. Dalong Wang in particular for their kind-hearted support. I would also like to thank all the academics of the CAS centre, and my friends Mr. Manjula Gunaratne, Mr. Sujeewa Fernando and Mr.Arjuna Dissanayake from Sri Lanka.

My abundant love must go to my parents for raising me and making my education a priority in their lives, for my loving wife Nadeeka sacrificing many things in her life to support my higher studies and to my loving new-born daughter Amaya for giving the added strength to successfully complete thesis work.

Lastly, I would like to dedicate this thesis to my parents, grandparents, all the good teachers who taught me from kindergarten to University entrance at St' Anthony's College, Kandy, and to all the lecturers who taught me at the undergraduate level at the Faculty of Engineering, University of Peradeniya, Sri Lanka for their encouragement to pursue postgraduate studies and to choose an academic career.

With Metha !

Asela K. Kulatunga
University of Technology, Sydney
Australia
29/08/2008

LIST OF PUBLICATIONS

Book chapters

1. **A.K. Kulatunga**, B. T. Skinner, D. K. Liu & H. T. Nguyen (2007), 'Simultaneous task allocation and motion coordination of autonomous vehicles using a parallel computing cluster'**,** *Robotic Welding, Intelligence and Automation,* Volume 362/2007, 409-420, Springer, Berlin/Heidelberg.

2. D.K. Liu &**A.K. Kulatunga**, (2007) 'Simultaneous Planning and Scheduling for Multi-Autonomous Vehicles', in Dahal, K., Tan, K.C. and Burke, E. (eds) *Evolutionary Scheduling, Studies in Computational Intelligence,* Volume 49/2007, 437-464, Springer, Berlin/Heidelberg.

Refereed Conference papers

3. **A. K. Kulatunga**, D. K. Liu & G. Dissanayake (2004) 'Simulated annealing algorithm based multi-robot coordination'. *Proceedings of the 3rd IFAC Symposium on Mechatronic Systems,* September 2004, Sydney, Australia, (Paper No. 74), 411-416

4. **A.K. Kulatunga**, D. K. Liu & S. B. Siyambalapitiya (2006) 'Ant colony optimization technique for simultaneous task allocation and path planning of autonomous vehicles.' *Proceedings of the IEEE International Conference on Cybernetics and Intelligent Systems (CIS),* 7-9 June, 2006, Bangkok, Thailand, 823-828

5. D.K. Liu, X. Wu, **A. K. Kulatunga**, G. Dissanayake (2006), 'Motion coordination of multiple autonomous vehicles in dynamic and strictly constrained environments.' *Proceedings of the IEEE International Conference on Cybernetics and Intelligent Systems (CIS),* 7-9 June, 2006, Bangkok, Thailand**,** 204-209

CONTENTS

ABSTRACT .... **III**
ACKNOWLEDGEMENT .... **V**
LIST OF PUBLICATIONS .... **VII**
CONTENTS .... **VIII**
LIST OF FIGURES .... **X**
LIST OF TABLES .... **XIII**
ABBREVIATIONS .... **XIV**
CHAPTER 1 .... **1**
INTRODUCTION **1**

1.1. BACKGROUND OF AUTONOMOUS VEHICLE OPERATIONS 1
1.2. PLANNING AND COORDINATION 3
1.3. SCOPE OF THE RESEARCH AND CONTRIBUTIONS 6
1.4. ORGANISATION OF THE THESIS 9

CHAPTER 2 .... **12**
LITERATURE SURVEY **12**

2.1. INTRODUCTION 12
2.2. INTEGRATED APPROACHES FOR MULTI-VEHICLE TASK ALLOCATION AND MOTION COORDINATION PROBLEM 13
2.2.1. EXACT APPROACHES 13
2.2.2. HEURISTIC / APPROXIMATION METHODS 14
2.3. TASK ALLOCATION FOR MULTIPLE AUTONOMOUS VEHICLES 17
2.4. VEHICLE ROUTING AND PATH / MOTION PLANNING 20
2.5. COLLISION AND DEADLOCK AVOIDANCE 23
2.6. RESEARCH AND DEVELOPMENT CHALLENGES 25
2.6.1. OVERALL EFFICIENCY AND SOLUTION QUALITY 25
2.6.2. OPTIMISATION METHODOLOGIES 26
2.6.3. PATH AND MOTION PLANNING ISSUES 27
2.7. SUMMARY 28

CHAPTER 3 .... **31**
PROBLEM FORMULATION AND SIMULTANEOUS PATH AND MOTION PLANNING ALGORITHM **31**

3.1. INTRODUCTION 31
3.2. TASK ALLOCATION AND MOTION COORDINATION PROBLEM 32
3.3. MOTION COORDINATION AND SIPAMOP ALGORITHM 34
3.4. SIMULATION ENVIRONMENT 41
3.5. SIMULATION STUDIES 43
3.5.1. COLLISION AVOIDANCE CAPABILITY 43
3.5.2. EFFICIENT MOTION COORDINATION CAPABILITY 50
3.6. CONCLUSION AND REMARKS 58

CHAPTER 4 .... **60**
SIMULTANEOUS TASK ALLOCATION AND MOTION COORDINATION - STATIC ENVIRONMENT **60**

4.1. INTRODUCTION 60
4.2. SIMULTANEOUS TASK ALLOCATION AND MOTION COORDINATION 62
4.3. MATHEMATICAL MODELLING 64
4.3.1. MATHEMATICAL MODEL 67
4.3.2. OPTIMISATION CRITERION 71

4.4. META-HEURISTIC ALGORITHMS FOR SIMULTANEOUS TASK ALLOCATION AND MOTION COORDINATION 74
4.4.1. SIMULATED ANNEALING ALGORITHM 74
4.4.2. ANT COLONY OPTIMISATION 77
4.4.3. AUCTION ALGORITHM 82
4.5. SIMULATION STUDIES 84
4.5.1. SIMULTANEOUS APPROACH VERSUS SEQUENTIAL APPROACH 85
4.5.2. COMPARISON OF THE SA, ACO AND AUCTION ALGORITHMS 92
4.6. DISCUSSION AND CONCLUSIONS 102

CHAPTER 5 105
STAMC APPROACH FOR A DYNAMIC ENVIRONMENT 105

5.1. INTRODUCTION 105
5.2. FORMULATION OF DYNAMIC MULTI-VEHICLE TASK ALLOCATION AND MOTION COORDINATION PROBLEM 109
5.2.1. MATHEMATICAL MODEL 109
5.3. THE DYNAMIC STAMC APPROACH 112
5.4. SIMULATION STUDIES AND RESULTS 114
5.4.1. SIMULATION STUDY 1 114
5.4.2. SIMULATION STUDY 2 117
5.4.3. RE-PLANNING DUE TO UNEXPECTED EVENTS (SIMULATION STUDY 3) 126
5.5. CONCLUSIONS AND DISCUSSION 131

CHAPTER 6 133
DISTRIBUTED IMPLEMENTATION OF THE STAMC APPROACH 133

6.1. INTRODUCTION 133
6.2. STAMC APPROACH IN DISTRIBUTED ENVIRONMENT 133
6.3. INTEGRATION OF MPITB IN THE MATLAB ENVIRONMENT 138
6.4. EXPERIMENT DESCRIPTION 142
6.4.1. SIMULATION PARAMETERS 142
6.4.2. CLUSTER COMPUTING ENVIRONMENT 143
6.5. RESULTS AND DISCUSSION 143
6.6. CONCLUSION AND FURTHER INVESTIGATIONS 146

CHAPTER 7 148
A CASE STUDY -APPLICATION OF THE STAMC APPROACH IN AN AUTOMATED CONTAINER TERMINAL 148

7.1. INTRODUCTION 148
7.2. REPRESENTATION OF THE AUTOMATED CONTAINER TERMINAL 149
7.3. CURRENT TASK ALLOCATION AND MOTION COORDINATION PROCESS 151
7.4. EXPERIMENTS WITH THE PROPOSED STAMC APPROACH 152
7.4.1 THE FIRST SIMULATION STUDY 153
7.4.2 SECOND SIMULATION STUDY 162
7.5 DISCUSSION 164

CHAPTER 8 166
CONCLUSION 166

8.1. INTRODUCTION 166
8.2. RESEARCH OUTCOMES 167
8.3. LIMITATIONS AND FUTURE OPPORTUNITIES FOR RESEARCH 168

REFERENCES 170

APPENDICES 181

List of Figures

Figure 1-1: Two possible approaches to solve the multiple-vehicle task allocation and motion coordination problem..5
Figure 1-2: Outline of the thesis 11
Figure 2-1: Organisation of literature survey..13
Figure 3-1: Schematic representation of the multi-vehicle task allocation and motion coordination problem with three key sub-problems 33
Figure 3-2: Schematic representation of simultaneous path and motion planning approach 36
Figure 3-3: The connections of nodes for the given example.....................................39
Figure 3-4: The flowchart of the SiPaMoP algorithm 41
Figure 3-5: The plan view of the simulation environment 42
Figure 3-6: The network map of the environment shown in Figure 3-5......................42
Figure 3-7: V1's and V2's Paths obtained without considering collisions by Dijkstra algorithm (Example 1) 44
Figure 3-8: V1's and V2's Paths obtained by the SiPaMoP algorithm: V2 wait till V1 passes node 34 to avoid collisions (Example 1) 45
Figure 3-9: V1's and V2's Paths obtained without considering collisions by Dijkstra algorithm (Example 2) 46
Figure 3-10: V1's and V2's Paths obtained by the SiPaMoP algorithm: By changing V2's path to avoid collisions between nodes 37 and 57 (Example 2) 47
Figure 3-11: Paths of all vehicles obtained by Dijkstra algorithm without considering collisions (Example 3) 49
Figure 3-12: Paths of all vehicles obtained by SiPaMoP algorithm by considering collisions (Example 3) 49
Figure 3-13: Vehicles V1 and V2 performing their first tasks (Path segments between node 110 -117 of V1 and from nodes 130 –137 of V2 are shown here).....................51
Figure 3-14: Vehicles V1 and V2 towards the completion of their task 1 (both use the same path segments from nodes 116 to node 95 and vehicle 2 travels behind the vehicle 1)..51
Figure 3-15: Vehicles V1 and V2 travelling to pick-up their 2nd tasks (at node 111 and 130 respectively) in a loop path topology 52
Figure 3-16: Vehicles V1 and V2 travelling to pick-up their 2nd tasks (Between loop segment 71 - 111 and 89 -131 respectively) 53
Figure 3-17: Vehicle 1 (from node 110 to 117) and Vehicle 2 (from node 130 to 60) perform their first tasks by following the paths planned by the SiPaMoP algorithm..54
Figure 3-18: Vehicle 1 (from node 115 to node 96) and vehicle 2 (from node 135 to node 60) perform their initial tasks planned by the SiPaMoP algorithm.....................55
Figure 3-19: Vehicle 1 returns to its 2nd task's origin while vehicle 2 is reaching its initial task's drop-off node (from the SiPaMoP algorithm)..55
Figure 3-20: Vehicle 1 travels towards its 2nd task's pick-up node of 111 while vehicle 2 travels towards its 2nd task's pick-up node of 131 56

Figure 3-21: Vehicles 1 and 2 performs their 2nd tasks by following the paths from the SiPaMoP algorithm 57
Figure 4-1: Schematic representation of the simultaneous approach 63
Figure 4-2: The simultaneous approach and the sequential approach 64
Figure 4-3: An example of tasks, vehicles’ start and drop-off nodes 66
Figure 4-4: Task allocation process: selecting appropriate task-vehicle pairs 69
Figure 4-5: Vehicle-task pair’s cost matrix 70
Figure 4-6: Flow chart of the Simulated Annealing Algorithm 77
Figure 4-7: Flow chart of the ACO algorithm 81
Figure 4-8: Simple Auction Process 83
Figure 4-9: Flow chart of Auction Algorithm 84
Figure 4-10: Flow chart of the simultaneous approach with the SA algorithm 88
Figure 4-11: Flow chart of the sequential approach with the SA algorithm 89
Figure 4-12: Variation of makespans obtained by simultaneous, sequential and SPS (without collision avoidance) respectively 91
Figure 4-13: Tasks allocation among vehicles, order of implementation and completion time obtained by the simultaneous approach for 8n-4m-case 2: makespan is 64.78 (stu) 92
Figure 4-14: Task allocation among vehicles, order of implementation and completion time obtained by the sequential approach for 8n-4-case 2: makespan is 72.29 (stu) ... 92
Figure 4-15: Gantt chart of the ES based the task allocation 95
Figure 4-16: Gantt chart of the ACO algorithm based task allocation 95
Figure 4-17: Task allocation results obtained from the SA algorithm 98
Figure 4-18: Task allocation results obtained from the ACO 99
Figure 4-19: Task allocation results obtained from AA 100
Figure 4-20: Summary of the simulation studies 103
Figure 5-1: Typical rescheduling methods 107
Figure 5-2: Flow chart of the priority based dynamic STAMC approach 113
Figure 5-3: Variation of makespan, tardiness and late tasks in simulation 1 115
Figure 5-4: Variation of makespan with re-scheduling intervals for different batch sizes 116
Figure 5-5: Variation of Tardiness with re-scheduling intervals for different batch sizes 116
Figure 5-6: Variation of number of late tasks with rescheduling intervals for different batch sizes 117
Figure 5-7: The completion time and tardiness of newly arrived tasks 119
Figure 5-8: Gantt chart of the initial schedule at Time =0 (stu) based on the DR 120
Figure 5-9: Gantt chart of initial schedule at Time = 0 (stu) based on AA 121
Figure 5-10: Gantt chart of initial schedule at Time = 0 (stu) based on SA algorithm 121
Figure 5-11: Gantt chart of 1st reschedule at Time =20 (stu) based on DR 122
Figure 5-12: Gantt chart of 1st reschedule at Time = 20 (stu) based on AA 123
Figure 5-13: Gantt chart of 1st reschedule at Time = 20 (stu) based on SA algorithm 123
Figure 5-14: Gantt chart of 2nd reschedule at Time = 40 (stu) based on DR 124
Figure 5-15: Gantt chart of 2nd reschedule at Time = 40 (stu) based on AA 125
Figure 5-16: Gantt chart of 2nd reschedule at Time = 40 (stu) based on SA algorithm 125

Figure 5-17: Schematic representation of the re-planning Strategy of the STAMC approach 127
Figure 5-18: Gantt chart of the schedule before the vehicle breakdown 128
Figure 5-19: Gantt chart after re-planning of the same example 129
Figure 5-20: Path representations before and after the breakdown 130
Figure 6-1: Flow diagram describing the simultaneous task allocation and motion coordination (STAMC) approach in parallel mode 135
Figure 6-2 : The data path of serial computation for the task allocation and motion coordination algorithm for autonomous vehicles 136
Figure 6-3 : The data path of parallel computation for the task allocation and motion coordination algorithm for autonomous vehicles. 137
Figure 6-4 : Software architecture showing the role of MPITB and other software components. 139
Figure 6-5 : Computation time (seconds) for the parallel/distributed and serial/centralised STAMC approach using 4 vehicles 144
Figure 6-6 : Computation time (seconds) for the parallel/distributed and serial/centralised STAMC approach using 6 vehicles 144
Figure 6-7 : Computation time (seconds) for the parallel/distributed and serial/centralised STAMC approach using 8 vehicles 145
Figure 6-8 : Computation time (seconds) for the parallel/distributed STAMC approach using 4/6/8 vehicles........ 146
Figure 7-1: Container terminal at Fisherman Island (http://www.patrick.com.au) ... 149
Figure 7-2: Arial view of the Fisherman Island Container terminal (www.googlemaps.com) 149
Figure 7-3: Different regions of the container yard at Fisherman's Island 150
Figure 7-4: Vehicle movements screen short of the MATLAB simulation platform 151
Figure 7-5: Results of the 1st simulation 154
Figure 7-6: Gantt chart of the 2nd hour schedule based on the AA 155
Figure 7-7: Gantt chart of the 2nd hour schedule based on the SA algorithm 155
Figure 7-8: Gantt chart of the 2nd hour schedule based on the ACO algorithm........ 156
Figure 7-9: Gantt chart of the 2nd hour schedule based on FCFS rule 156
Figure 7-10: Gantt chart of the 2nd hour schedule based on COF rule 157
Figure 7-11: Gantt chart of the 4th hour schedule based on the AA 157
Figure 7-12: Gantt chart of the 4th hour schedule based on the SA algorithm 158
Figure 7-13: Gantt chart of the 4th hour schedule based on the ACO algorithm 158
Figure 7-14: Gantt chart of the 4th hour schedule based on COF rule 159
Figure 7-15: Gantt chart of the 4th hour schedule based on FCFS rule 159
Figure 7-16: Gantt chart of the 6th hour schedule based on the AA 160
Figure 7-17: Gantt chart of the 6th hour schedule based on the SA algorithm 160
Figure 7-18: Gantt chart of the 6th hour schedule based on the ACO algorithm 161
Figure 7-19: Gantt chart of the 6th hour schedule based on COF rule 161
Figure 7-20: Gantt chart of the 6th hour schedule based on FCFS rule 162
Figure 7-21: Gantt chart of overall schedule for 8 hours based on FCFS rule 163
Figure 7-22: Gantt chart of overall schedule for 8 hours based on COF rule 164

List of Tables

Table 3-1: Path selection of example 1(One vehicle waits till the other one passes away the connection) 44
Table 3-2: Path selection of the vehicles in example 2 (V2 changes its path to avoid collision) 47
Table 3-3: Task allocation information to four vehicles of example 3 48
Table 3-4: Task allocation information and path details in loop based path topology 50
Table 3-5: Vehicles task allocation information and path details in conventional path topology 53
Table 3-6: Completion and empty travel times obtained from the two approaches 57
Table 4-1: Different simulation problem sizes and makespan values 91
Table 4-2: Makespan comparison of ACO algorithm with optimal Value 93
Table 4-3: Empty travel times (stu) of 8T-4m-case 1 94
Table 4-4: Comparison of makespan, CPU time and empty travel times of ACO and SA algorithms 96
Table 4-5: Eight tasks' pick-up and drop-off nodes 97
Table 4-6: Four vehicles' initial positions 97
Table 4-7: Simulation results obtained by SA algorithm based STAMC approach 97
Table 4-8: Simulation results obtained by ACO based STAMC approach 98
Table 4-9: Simulation results obtained by AA based STAMC approach 99
Table 4-10: Comparison of Makespan and vehicle utilisation of three methods 100
Table 4-11: Makespan comparisons of ACO, SA and Auction algorithms 101
Table 4-12: Hardware and Software Specification of Simulation Studies 102
Table 4-13: Algorithms and parameter values 102
Table 5-1: Variation of rescheduling intervals with tardiness, late tasks and tasks scheduled 115
Table 5-2: Task allocation among the four vehicles before and after vehicle 4 breaks down 129
Table 6-1 : Algorithm and simulation parameters 142
Table 6-2 : Cluster computing hardware and software environment 143
Table 7-1: Task allocation information of the existing method of one hour duration 152
Table 7-2: The makespan and computational cost of the first scenario 154

Abbreviations

ACO	Ant colony optimization
AGV	Automated guided vehicle
AV	Autonomous vehicle
BD	Breakdown
BS	Batch size
BSA	Beam Search Algorithm
CIM	Computer Integrated Manufacturing
COF	Close proximity task first
CR	Cooling rate
CT	Container Terminals
DR	Dispatching Rules
FCFS	First-Come-First-Served
FMS	Flexible Manufacturing Systems
GA	Genetic algorithm
LAN	Local area network
MC	Motion Coordination
MPI	Message-Passing Interface
MS	Makespan
PP	Path Planning
RSI	Rescheduling interval
SA	Simulated Annealing
SCs	straddle carriers
SiPaMoP	Simultaneous Path and Motion Planning
SPS	Shortest Path Search
STAMC	Simultaneous Task Allocation and Motion Coordination
stu	simulation time units
TA	Task Allocation
TEU	Twenty feet Equivalent Units
TS	Tabu Search